WE CLAIM

1. A data processing apparatus, comprising:

a processor operable in a plurality of modes and a plurality of domains, said plurality of domains comprising a secure domain and a non-secure domain, said plurality of modes including at least one non-secure mode being a mode in the non-secure domain, and at least one secure mode being a mode in the secure domain, said processor being operable such that when executing a program in a secure mode said program has access to secure data which is not accessible when said processor is operating in a non-secure mode;

a memory operable to store data required by the processor and comprising secure memory for storing secure data and non-secure memory for storing non-secure data, the processor being operable to issue a memory access request when access to an item of data in the memory is required;

at least one memory management unit operable, upon receipt of the memory access request from the processor, to perform conversion of a virtual address specified by the memory access request to a physical address;

a first set of tables, each table in the first set containing a number of first descriptors, each first descriptor containing at least a virtual address portion and a corresponding intermediate address portion;

a second set of tables, each table in the second set containing a number of second descriptors, each second descriptor containing at least an intermediate address portion and a corresponding physical address portion, the second set of tables being managed by the processor when operating in a privileged mode which is not a non-secure mode;

the at least one memory management unit being operable to cause predetermined tables in said first and second set to be referenced to enable the conversion of the virtual address specified by the memory access request to a physical address.

2. A data processing apparatus as claimed in Claim 1, wherein the privileged mode is a monitor mode in which the processor is operable to manage switching between said secure domain and said non-secure domain.

3. A data processing apparatus as claimed in Claim 1, wherein said privileged mode is a privileged secure mode.

4. A data processing apparatus as claimed in Claim 3, wherein in said at least one non-secure mode the processor is operable under the control of a non-secure operating system and in said at least one secure mode the processor is operable under the control of a secure operating system, the secure operating system being operable to manage the second set of tables when the processor is operating in the privileged secure mode.

5. A data processing apparatus as claimed in claim 2, wherein when switching between the secure domain and the non-secure domain, the processor is operable in the monitor mode to select the predetermined tables in said first and second sets, dependent on whether the domain being switched to is the secure domain or the non-secure domain.

6. A data processing apparatus as claimed in Claim 1, wherein the predetermined tables within said first and second sets are selected when the tables are to be referenced dependent on whether the processor is operating in a secure mode or a non-secure mode at the time the memory access request is issued.

7. A data processing apparatus as claimed in Claim 1, wherein the at least one memory management unit comprises a first memory management unit and a second memory management unit.

8. A data processing apparatus as claimed in Claim 7, wherein the tables in the first set are associated with the first memory management unit and the tables in the second set are associated with the second memory management unit.

9. A data processing apparatus as claimed in Claim 7, wherein if the first memory management unit needs to access a first descriptor within a predetermined table of said first set, it issues a table lookup request specifying an intermediate address for that first descriptor, the second memory management unit being operable to receive the table

lookup request and determine the physical address corresponding to that intermediate address.

10. A data processing apparatus as claimed in Claim 9, wherein the second memory management unit is then operable to cause the first descriptor at that physical address to be retrieved and returned to the first memory management unit.

11. A data processing apparatus as claimed in Claim 7, wherein the first memory management unit comprises a first internal storage unit for storing first descriptors retrieved from the predetermined table of the first set, and used by the first memory management unit to derive access control information used to perform the conversion of the virtual address into a corresponding intermediate address.

12. A data processing apparatus as claimed in Claim 11, wherein the first internal storage unit is a first translation lookaside buffer (TLB) operable to store the first descriptors retrieved from the predetermined table of the first set.

13. A data processing apparatus as claimed in claim 12, wherein the first TLB is a first main TLB for storing the first descriptors retrieved by the first memory management unit from the predetermined table of the first set, and the internal storage further comprises a micro-TLB for storing the access control information derived from the first descriptors, the access control information comprising conversions between a number of virtual address portions and corresponding intermediate address portions, and the access control information being transferred from the first main TLB to the micro-TLB prior to use of that access control information by the first memory management unit.

14. A data processing apparatus as claimed in Claim 7, wherein the first memory management unit comprises a first internal storage unit for storing new descriptors derived from corresponding first and second descriptors retrieved from the predetermined tables of the first and second sets, and used by the first memory management unit to derive access control information used to perform the conversion of the virtual address into a corresponding physical address.

15. A data processing apparatus as claimed in Claim 14, wherein the first internal storage unit is a first translation lookaside buffer (TLB) operable to store the new descriptors derived from corresponding first and second descriptors.

16. A data processing apparatus as claimed in claim 15, wherein the first TLB is a first main TLB for storing the new descriptors derived from corresponding first and second descriptors, and the internal storage unit further comprises a micro-TLB for storing the access control information, the access control information comprising conversions between a number of virtual address portions and corresponding physical address portions, and the access control information being transferred from the first main TLB to the micro-TLB prior to use of that access control information by the first memory management unit.

17. A data processing apparatus as claimed in Claim 7, wherein the second memory management unit comprises a second internal storage unit for storing second descriptors retrieved from the predetermined table of the second set, and used by the second memory management unit to derive access control information used to perform the conversion of the intermediate address into a corresponding physical address.

18. A data processing apparatus as claimed in Claim 17, wherein the second internal storage unit is a second translation lookaside buffer (TLB) operable to store the second descriptors retrieved from the predetermined table of the second set.

19. A data processing apparatus as claimed in claim 18, wherein the second TLB is a second main TLB for storing the second descriptors retrieved by the second memory management unit from the predetermined table of the second set, and the internal storage further comprises a micro-TLB for storing the access control information derived from the second descriptors, the access control information comprising conversions between a number of intermediate address portions and corresponding physical address portions, and the access control information being transferred from the second main TLB to the

micro-TLB prior to use of that access control information by the second memory management unit.

20. A data processing apparatus as claimed in Claim 18, wherein the first internal storage unit is a first translation lookaside buffer (TLB) operable to store the first descriptors retrieved from the predetermined table of the first set, and wherein the first and second sets of tables each comprise at least a secure table and a non-secure table, the first and second TLBs comprising a flag associated with each descriptor stored therein to identify whether that descriptor is derived from said non-secure table or said secure table.

21. A data processing apparatus as claimed in Claim 19, wherein the first and second sets of tables each comprise at least a secure table and a non-secure table, the first and second TLBs comprising a flag associated with each descriptor stored therein to identify whether that descriptor is derived from said non-secure table or said secure table, and wherein the micro-TLB of both the first and second memory management units is flushed whenever the mode of operation of the processor changes between a secure mode and a non-secure mode, in the secure mode access control information only being transferred to the micro-TLB from a descriptor in the associated first or second main TLB that said associated flag indicates is from the secure table, and in the non-secure mode access control information only being transferred to the micro-TLB from a descriptor in the associated first or second main TLB that said associated flag indicates is from the non-secure table.

22. A data processing apparatus as claimed Claim 1, wherein the at least one memory management unit comprises a single memory management unit, and the processor is operable to execute table merging code to reference the predetermined tables of the first and second sets in order to produce from a first descriptor and an associated second descriptor a new descriptor associating a virtual address portion with a corresponding physical address portion.

23. A data processing apparatus as claimed in Claim 22, wherein the table merging code is operable to retrieve the first descriptor after referencing the predetermined table in the second set to obtain the physical address of the first descriptor.

24. A data processing apparatus as claimed in Claim 23, wherein the table merging code is operable to use the first descriptor to determine the intermediate address corresponding to the virtual address specified by the memory access request, and to then reference the predetermined table in the second set to obtain the second descriptor providing a physical address for that intermediate address, whereafter the table merging code is operable to merge the first and second descriptors to produce the new descriptor.

25. A data processing apparatus as claimed in Claim 22, wherein the single memory management unit comprises an internal storage unit for storing the new descriptor produced by the table merging code, and used by the single memory management unit to derive access control information used to perform the conversion of the virtual address into a corresponding physical address.

26. A data processing apparatus as claimed in Claim 25, wherein the processor is operable to execute the table merging code when the access control information required to determine the physical address for the memory access request is not found in the internal storage unit.

27. A data processing apparatus as claimed in Claim 25, wherein the internal storage unit is a translation lookaside buffer (TLB) operable to store the new descriptors produced by the table merging code.

28. A data processing apparatus as claimed in claim 27, wherein the TLB is a main TLB for storing the new descriptors obtained by the single memory management unit from the table merging code, and the internal storage further comprises a micro-TLB for storing the access control information derived from the new descriptors, the access control information comprising conversions between a number of virtual address portions and corresponding physical address portions, and the access control information being

transferred from the main TLB to the micro-TLB prior to use of that access control information by the single memory management unit.

29. A data processing apparatus as claimed in Claim 27, wherein the first and second sets of tables each comprise at least a secure table and a non-secure table, the TLB comprising a flag associated with each new descriptor stored therein to identify whether that new descriptor is derived from said non-secure tables or said secure tables.

30. A data processing apparatus as claimed in Claim 28, wherein the first and second sets of tables each comprise at least a secure table and a non-secure table, the TLB comprising a flag associated with each new descriptor stored therein to identify whether that new descriptor is derived from said non-secure tables or said secure tables, and wherein the micro-TLB of the single memory management unit is flushed whenever the mode of operation of the processor changes between a secure mode and a non-secure mode, in the secure mode access control information only being transferred to the micro-TLB from a new descriptor in the main TLB that said associated flag indicates is derived from secure tables, and in the non-secure mode access control information only being transferred to the micro-TLB from a new descriptor in the main TLB that said associated flag indicates is derived from non-secure tables.

31. A data processing apparatus as claimed in Claim 22, wherein the privileged mode is a monitor mode in which the processor is operable to manage switching between said secure domain and said non-secure domain, and wherein the table merging code is executed by the processor when operating in the monitor mode.

32. A data processing apparatus as claimed in Claim 1, wherein said first and second sets of tables comprise page tables.

33. A data processing apparatus as claimed in Claim 1, wherein the first set of tables and the second set of tables are stored within said memory.

34. A method of controlling access to a memory in a data processing apparatus, the data processing apparatus comprising a processor operable in a plurality of modes and a plurality of domains, said plurality of domains comprising a secure domain and a non-secure domain, said plurality of modes including at least one non-secure mode being a mode in the non-secure domain, and at least one secure mode being a mode in the secure domain, said processor being operable such that when executing a program in a secure mode said program has access to secure data which is not accessible when said processor is operating in a non-secure mode, the memory being operable to store data required by the processor and comprising secure memory for storing secure data and non-secure memory for storing non-secure data, the method comprising the steps of:

providing a first set of tables, each table in the first set containing a number of first descriptors, each first descriptor containing at least a virtual address portion and a corresponding intermediate address portion;

providing a second set of tables, each table in the second set containing a number of second descriptors, each second descriptor containing at least an intermediate address portion and a corresponding physical address portion, the second set of tables being managed by the processor when operating in a privileged mode which is not a non-secure mode;

issuing from the processor a memory access request when access to an item of data in the memory is required; and

performing conversion of a virtual address specified by the memory access request to a physical address with reference to predetermined tables in said first and second set.

35. A method as claimed in Claim 34, wherein the privileged mode is a monitor mode in which the processor is operable to manage switching between said secure domain and said non-secure domain.

36. A method as claimed in Claim 34, wherein said privileged mode is a privileged secure mode.

37. A method as claimed in Claim 36, wherein in said at least one non-secure mode the processor is operable under the control of a non-secure operating system and in said at least one secure mode the processor is operable under the control of a secure operating system, the secure operating system being operable to manage the second set of tables when the processor is operating in the privileged secure mode.

38. A method as claimed in claim 35, wherein when switching between the secure domain and the non-secure domain, the processor is operable in the monitor mode to select the predetermined tables in said first and second sets, dependent on whether the domain being switched to is the secure domain or the non-secure domain.

39. A method as claimed in claim 34, wherein the predetermined tables within said first and second sets are selected when the tables are to be referenced dependent on whether the processor is operating in a secure mode or a non-secure mode at the time the memory access request is issued.

40. A method as claimed in claim 34, wherein said step of performing conversion of a virtual address to a physical address is performed by at least one of a first memory management unit and a second memory management unit.

41. A method as claimed in Claim 40, wherein if the first memory management unit needs to access a first descriptor within a predetermined table of said first set, the method further comprises the steps of:

issuing from the first memory management unit a table lookup request specifying an intermediate address for that first descriptor; and

receiving the table lookup request at the second memory management unit and determining the physical address corresponding to that intermediate address.

42. A method as claimed in Claim 41, further comprising the step of:

causing the first descriptor at that physical address to be retrieved and returned to the first memory management unit.

43. A method as claimed in Claim 42, further comprising the steps of:

causing a second descriptor within a predetermined table of said second set to be retrieved; and

merging the first descriptor and second descriptor in order to produce a new descriptor for storing in the first memory management unit, the new descriptor containing at least a virtual address portion and a corresponding physical address portion.

44. A method as claimed in claim 34, wherein the data processing apparatus comprises a single memory management unit, and the method comprises the step of:

executing table merging code to reference the predetermined tables of the first and second sets in order to produce from a first descriptor and an associated second descriptor a new descriptor associating a virtual address portion with a corresponding physical address portion.

45. A method as claimed in Claim 44, wherein the table merging code is operable to perform the steps of:

referencing the predetermined table in the second set to obtain the physical address of the first descriptor; and

retrieving the first descriptor.

46. A method as claimed in Claim 45, wherein the table merging code is further operable to perform the steps of:

using the first descriptor to determine the intermediate address corresponding to the virtual address specified by the memory access request;

referencing the predetermined table in the second set to obtain the second descriptor providing a physical address for that intermediate address; and

merging the first and second descriptors to produce the new descriptor.

47. A method as claimed in claim 44, wherein the single memory management unit comprises an internal storage unit for storing access control information derived from the new descriptor produced by the table merging code, and used by the single memory

management unit to perform the conversion of the virtual address into a corresponding physical address.

48. A method as claimed in Claim 47, wherein the processor is operable to execute the table merging code when the access control information required to determine the physical address for the memory access request is not found in the internal storage unit.

49. A method as claimed in claim 44, wherein the privileged mode is a monitor mode in which the processor is operable to manage switching between said secure domain and said non-secure domain, and wherein the table merging code is executed by the processor when operating in the monitor mode.

50. A computer program providing table merging code and operable to configure a processor of a data processing apparatus to perform the method of claim 44.

51. A computer program product carrying a computer program as claimed in Claim 50.